

Navigate to Remember: A Declarative Memory Model for Incremental Semantic Mapping

Wei Hong Chin¹, Naoyuki Kubota¹, Zhaojie Ju², and Honghai Liu²

¹ Faculty of Systems Design, Tokyo Metropolitan University, Hino, Japan
chin-wei hong@ed.tmu.ac.jp, kubota@tmu.ac.jp

² University of Portsmouth, Portsmouth, UK
zhaojie.ju, honghai.liu@port.ac.uk

Abstract. Biologically inspired computational techniques play a crucial role in robotic cognition. Artificial learning agents and robots that interact in complex environments must constantly acquire and refine knowledge over long periods of time. In this paper, we propose a novel recurrent neural architecture that mimics humans' declarative memory system for continuously generating a cognitive map during robot navigation. The proposed method termed as Declarative Memory Adaptive Recurrent Model (DM-ARM), and consists of three hierarchical memory courses: i) Working Memory, ii) Episodic Memory and iii) Semantic Memory layer. Each memory layer comprises a self-organizing adaptive recurrent incremental network (SOARIN) with a different learning task respectively. The Working Memory layer quickly clusters sensory information while the Episodic Memory layer learns fine-grained spatiotemporal relationships of clusters (temporal encoding). Both the memory layer learning is in an unsupervised manner. The Semantic Memory layer utilizes task-relevant cues to adjust the level of architectural flexibility and generate a semantic map that contains more compact episodic representations. The effectiveness of the proposed recurrent neural architecture is evaluated through a series of experiments. We implemented and validated our proposed work on the tasks of robot navigation.

Keywords: cognitive map, navigation, mobile robot, episodic memory

1 Introduction

Spatial cognition is the basic ability of mammals to map, locate and navigate in different environments [1]. However, up-to-date intelligence computational models are still far from creating an artificial agent that can accomplish daily chores in an unstructured environment. This is a common task which requires semantic information and episodic memory [2].

The semantic memory is known as the brain mechanism for acquiring and constructing the internal spatial representation. In the meantime, episodic memory allows people to learn higher-level tasks through self-experience and to plan actions respectively. In order to perform spatiotemporal tasks, both elements

play crucial roles for humans. In both cognitive science and neuroscience, extensive research has been carried out to reveal the semantic memory and episodic memory's fundamental principles and neural bases [3–5].

The cognitive map is a kind of intelligence description of the external environment since animal cognition of the environment is an expression of the external environment. It represents the relative geometric relationship of the external space features. At present, cognitive map-based robot navigation has drawn the attention of many researchers. Tian, et al. [6] proposed a brain-inspired SLAM to build a cognitive map of the office environment by combining odometry information with RGB-D sensors data. The cognitive map consists of a set of space coordinates that the robot has experienced and these nodes used to generate a global path. Chin et al. [8] proposed an unsupervised learning model of episodic memory to categorize and encode experiences of a robot to the environment and generates a cognitive map.

Recently, the Complementary Learning System (CLS) theory has been updated to consolidate additional findings of neuroscience [9, 10]. The first set of findings considers the role of memory replay stored in the hippocampus as a mechanism that supports the goal-oriented manipulation of experience statistics as well as the integration of new information. The hippocampus promptly encodes episodic events that can be reactivated during resting and intentional or unintentional memory recall. In this way, the information in the neocortex is consolidated by reactivating encoded experiences in terms of multiple internal replays. The combination of working memory, episodic memory, and semantic memory are known as declarative memory.

In this paper, we propose a novel recurrent neural architecture that mimics humans' declarative memory system to build a semantic map incrementally for robot navigation. The proposed architecture consists of three hierarchical memory layers: i) Working Memory, ii) Episodic Memory and iii) Semantic Memory layer. Each memory layer comprises a self-organizing adaptive recurrent incremental network (SOARIN) with a different learning task respectively. The Working Memory layer quickly learns feature vectors from sensors (clustering); while the Episodic Memory layer learns fine-grained spatiotemporal relationships of feature vectors (temporal encoding). Both the memory layer learning is in an unsupervised manner. The Semantic Memory layer utilizes task-relevant cues to adjust the level of architectural flexibility and generate a semantic topological map (categorization). For the navigation task, we used text-to-speech to pass information to the robot for space labeling in Semantic Memory layer. All networks in the respective layer can grow or shrink for adapting incoming sensory information. In order to alleviate the catastrophic forgetting, the Episodic Memory layer regularly reactivates previously learned temporal neurons activations and replays to itself and to the Semantic Memory layer (memory replay) during robot rest time. The memory replay is accomplished without the needs of external sensory information. The memory replay mechanism allows the robot to continually learn incoming novel sensory input while retaining knowledge that has been learned previously.

Contributions of this paper are i) the proposed method, DM-ARM enables a robot to build a semantic topological map from scratch and continually update the map for global path planning and navigation; ii) DM-ARM overcomes the catastrophic forgetting which helps the robot *remembers* previously explored environment while moving to new environments; and iii) the robot utilizes the generated semantic topological map to switch its moving behaviors according to environmental conditions such as fall-following, obstacle avoidance, and fast-speed moving.

2 Proposed Method: DM-ARM

DM-ARM architecture consists of three hierarchical Self-organizing Adaptive Recurrent Incremental Network (SOARIN) as shown in Figure. SOARIN is the evolved version of the Gamma-GWR [11] that adaptively and dynamically generate new neurons and topology connections according to novel consecutive sensory input. The Working Memory layer (WML) instantly clusters incoming input vectors in an unsupervised manner. Next, the episodic memory layer (EML) receives firing neuron weights from WML and encodes the spatiotemporal neurons activation patterns. Next, the Semantic Memory layer (SML) receives neural activation trajectories from EML and task-relevant signals (labels) from users to update the network and generate more compact representations of episodic experience with semantic meaning. Therefore, WML, EML, and SML mitigate catastrophic forgetting through continually generate neurons if novel input is encountered, update neurons weights if received input is similar with previously learned knowledge. In addition, neurons that are inactive for a long time will be eliminated automatically for maintaining the storage capacity.

In robot navigation, WML works as a novelty detector where each neuron in the network represents a group of similar input vectors and generate new neurons if incoming input vectors do not fit into the network. In EML, the network encodes the sequence of the robot's movement and each neuron in the network stores the robot's location for local localization. In SML, the network encodes a set of neurons in EML to define a place of the explored environment. Each neuron in SML represents an area of the environment and the robot utilizes the information for switching its moving behaviors such as wall following, obstacle avoidance or fast-speed moving. The overall architecture as shown as Figure 1

2.1 Working Memory Layer

Each memory layer contains a Self-Organizing Adaptive Recurrent Incremental Network (SOARIN) which is an adaptive recurrent extension of the Gamma-Grow-When-Required-Network (Gamma-GWR) [11] self-organizing network. The SOARIN embeds a self-adaptive learning threshold that allows the network to dynamically grow or shrink according to input vectors. SOARIN serves as the short-term memory buffer in which new neurons will be generated to represent incoming inputs and connections will be generated to link neurons that activated

4 Wei Hong Chin et al.

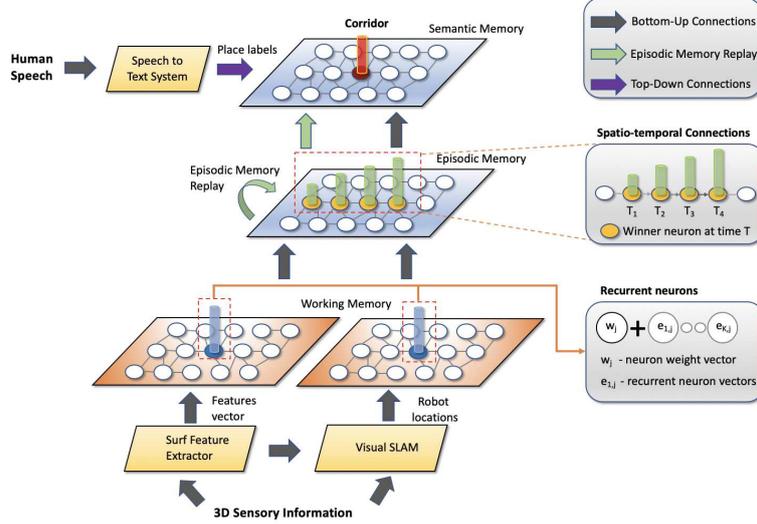


Fig. 1. DM-ARM overall architecture

subsequently via competitive Hebbian learning. The notations of the SOARIN is tabulated in Table 1.

Initially, the network generates 2 recurrent neurons based on received sensory inputs. Each neuron of the layer consists of a weight vector w_j and a number K of temporal attributes e_j^k . For subsequent learning, the network computes the neuron that best matches with the current sensory $x(t)$ using equation 1,2 and 3.

$$b = \arg \min(T_j), \quad (1)$$

$$T_j = \alpha_0 \|x(t) - w_j\|^2 + \sum_{k=1}^K \alpha_k \|E_k(t) - e_{j,k}\|^2, \quad (2)$$

$$E_k(t) = \beta \cdot w_b(t-1) + (1 - \beta) \cdot e_{b,k-1}(t-1) \quad (3)$$

Next, the activation value of the best matching neuron b is computed as follows:

$$a_b(t) = \exp(-T_b) \quad (4)$$

If the activation value $a_b(t)$ less than a preset threshold a_T . A new neuron N is added to the network with the new weights as below:

$$w_N = 0.5 \cdot (x(t) + w_b) \quad (5)$$

Table 1. The SOARIN notations

Notation	Definition
k	Number of temporal attribute
T_j	Activation value of neuron j
$w_b(t-1)$	Best matching neuron weights $t-1$
$E_k(t)$	Global attribute of the network at t , $E_k(t=0) = 0$
$e_{j,k}(t)$	Temporal elements of neuron j
$\alpha_i, \beta \in [0, 1]$	Contributing factors
r_j	Regularity counter of neuron j
τ_j, λ	Decay factors for regularity counter
ρ	Learning threshold
$P_{(m,n)}$	Temporal connection between neuron m and n
V	Associative matrix for labeling
b	Index of best matching neuron

$$e_{k,N} = 0.5 \cdot (E_k(t) + e_{k,b}) \quad (6)$$

A new connection is created to connect the neuron b and the second best matching neuron. If the $a_b(t)$ larger than a_T , it means that the neuron b can represents the input $x(t)$. Thus, the neuron b and its neighbor neurons n is updated according to input $x(t)$ as follows:

$$w_{j(\text{new})} = \gamma_j \cdot r_j \cdot (x(t) - w_{j(\text{old})}) \quad (7)$$

$$e_{j,k(\text{new})} = \gamma_j \cdot r_j \cdot (E_k(t) - e_{j,k(\text{old})}) \quad (8)$$

If there is no connection between best matching neuron $a_b(t)$ and second best matching neuron, a new connection will be created to connect them. Each edge has an age counter that increases by one at each iteration. The age of the connection between best matching neuron and second best matching neuron is reset to zero. Connections with an age larger than a preset threshold will be removed, and neurons without connections are deleted from the network.

We introduce a self-update threshold that allows the threshold to adjust its value adaptively in response to sensory input. The self-adjust equation as follows:

$$\rho_{(\text{new})} = (1 - r_b) \cdot a_b(t) + r_b \cdot \rho_{(\text{old})} \quad (9)$$

The self-update equation adjusts the threshold value that closes to the best matching neuron activation value. This means that the threshold value is changing dynamically. This is because the robot observes the environment sequentially,

6 Wei Hong Chin et al.

based on the nature of SOARIN learning, it is able to recognize the sequential sensory data and lower the threshold allows the network to update itself without adding new neurons. Thus, it helps to overcome the node proliferation issue. However, the threshold is reset to the baseline if a mismatch occurs in the network. The best matching neuron's weight are feed forward to the episodic memory layer as input vector.

2.2 Episodic Memory Layer

In episodic memory layer, the network learning is similar to WML with additional conditions. First, in each iteration, the input of the network is WML best matching neuron's weight. Next, each episodic neuron contains a regularity counter $r_j \in [0, 1]$ indicating its firing strength over time. Newly generated episodic neuron has value of $r_j = 1$. In each iteration, the regularity value of best matching neuron and its neighbor neurons decrease using equation as follows:

$$\Delta r_j = \tau_j \cdot \lambda \cdot (1 - r_j) - \tau_j \quad (10)$$

In addition, a new episodic neuron will only be added to the network if $a_b^{\text{em}}(t) < \rho_a^{\text{em}}$ and $r_b^{\text{em}} < \rho_r^{\text{em}}$. Thus, the episodic network resets its threshold value and sends a feedback signal back to WML to reset its threshold ρ_a^{wm} is send. If the activation and regularity value are fulfilled the threshold, the episodic neurons update using equation 7 and 8.

In episodic memory structure, a sequence of events forms an episode to store specific past experiences and episodes are correlating to each other. We implement temporal connections that learn activation patterns of recurrent neurons in the network.

The temporal connections encode the sequence of neurons that have been activated during the learning stage. For each learning iteration, a temporal link will be increased by 1 between two neurons that are sequentially stimulated. Specifically, when best matching neuron b that triggered at time t and $t - 1$ subsequently, the temporal link between them is reinforced as follow:

$$P_{(b(t), b(t-1))}^{\text{new}} = P_{(b(t), b(t-1))}^{\text{old}} + 1, \quad (11)$$

In this way, for each recurrent neuron m , the next neuron g can be retrieved from the encoded temporal sequence by choosing the largest value of P as below:

$$g = \arg \max P_{(m, n)} \quad (12)$$

where n are the neighbors of m . As a result, the activation sequence of recurrent neurons can be restored without requiring any input data.

In order to generate meaningful sequential data for the playback purpose, we utilize the spatiotemporal relationship of neurons that encoded in the episodic memory layer. The sequential data playback can be generated in episodic memory layer for each episodic neuron whenever the network receives incoming sensory data. For example, if the winner episodic neuron b is triggered by input

data, we can determine the next temporal neuron by choosing the neuron that has the largest activation value of P . A set of neurons playback with length $K^{\text{sm}} + K^{\text{em}} + 1$ for each neuron j is computed as follows:

$$U_j = \langle w_{u(0)}^{\text{em}}, w_{u(1)}^{\text{em}}, \dots, w_{u(K^s)}^{\text{em}} \rangle \quad (13)$$

$$u(i) = \arg \max P_{(j, u(i-1))} \quad (14)$$

where $P(i, j)$ is the episodic temporal connection matrix and $s(0) = j$.

2.3 Semantic Memory layer

The semantic memory layer is hierarchically connecting with episodic memory layer. It consists of an ar-GWR network which receives bottom-up inputs from the episodic memory layer, top-down inputs such as labels or tags for developing representations that contain semantic knowledge across a wider temporal scale. The semantic knowledge can be retrieved by providing cues from the top-down signals.

The neural activities mechanism in the semantic memory layer is similar to the episodic memory layer with an additional condition for generating a new neuron. In this layer, neuron learning happens if the network accurately predicts the class label of a labeled input sequence from the episodic memory layer through the learning process. A new neuron will be created if the predicted class label from the network is incorrect. Thus, this is the additional factor that modulates the neurons update rate. In addition, each semantic neuron encodes information over higher temporal sequences than episodic neurons due to the hierarchical learning of input data.

Episodic memory is formed by a sequence of activation events. In previous work, authors only calculate the distance difference of the semantic best matching neuron's weight and bottom-up episodic inputs for semantic learning without measuring the inter-relationship of events between them. The inter-relationship of episode neurons is important for recognition and retrieval purpose. Therefore, we introduce a new equation to measure the inter-relationship between the best matching neuron's weight and bottom-up inputs for improving the semantic learning performance. Thus, Equation 1-3 becomes:

$$b_s = \arg \min(T_j^{\text{sm}}), \quad (15)$$

8 Wei Hong Chin et al.

$$\begin{aligned}
T_j^{\text{sm}} = & \alpha_1 \| w_b^{\text{em}} - w_j^{\text{sm}} \|^2 + \\
& \sum_{k=1}^K \alpha_k^{\text{sm}} \| E_k^{\text{sm}}(t) - e_{k,j}^{\text{sm}}(t) \|^2 \cdot \\
& \exp \left(- \left(1 - \frac{w_b^{\text{em}} \cdot w_j^{\text{sm}}}{\| w_j^{\text{sm}} \| \cdot \| w_b^{\text{em}} \|} + \right. \right. \\
& \left. \left. \sum_{k=1}^K \left(\frac{E_k^{\text{sm}}(t) \cdot e_{k,j}^{\text{sm}}(t)}{\| e_{k,j}^{\text{sm}} \| \cdot \| E_k^{\text{sm}}(t) \|} \right) \right) \right), \quad (16)
\end{aligned}$$

$$E_k^{\text{sm}}(t) = \beta \cdot w_{j-1}^{\text{sm}} + (1 - \beta) \cdot e_{k-1,j-1}^{\text{sm}} \quad (17)$$

With these equations, the selected neuron either is expected to be the correct semantic neuron for the particular sequence of episodic inputs or it is more notable than other semantic neurons or both.

The semantic memory layer receives input neural data from the episodic memory layer which means the BMNs of the episodic memory layer with respect to $x(t)$. Hence, BMNs in this layer are calculated using equations 15-17. Note that, the input is from bottom-up neural episodic weights, therefore $x(t)$ is substituted by w_b^{em} for the network learning. The labeling method is similar to episodic memory layer where each neuron in semantic memory layer is assigned to a label that obtains from $x(t)$ using equation 12 and ??.

As a result, a new semantic neuron is generated only if the BMN b does not fulfill 3 conditions: 1) $a_b^{\text{sm}}(t) < \rho_a$; 2) $r_b^{\text{sm}} < \rho_r$; and 3) BMN's label ζ_b^{sm} is not identical with the data input's label ζ (equation ??). Note that, this label matching condition in semantic memory layer is not taken into account if the data input is not labeled. If the winner semantic neuron b predicts the label ζ_b that same with the class label ζ of input $x(t)$, then the neuron learning process is triggered with an additional learning factor $\psi = 0.001$. Therefore, Equation 7 and 8 become:

$$w_{j(\text{new})}^{\text{sm}} = \psi \cdot \gamma_j \cdot r_j \cdot (w_b^{\text{em}} - w_{j(\text{old})}^{\text{sm}}), \quad (18)$$

$$e_{k(\text{new})}^{\text{sm}} = \psi \cdot \gamma_j \cdot r_j \cdot (E_k^{\text{sm}}(t) - e_{k,j(\text{old})}^{\text{sm}}) \quad (19)$$

As such, the semantic memory layer learns more compact representations with respect to data input labels. The data labels control the layer stability and plasticity where new semantic neurons are generated only if the network is unable to predict the correct class label of data input. In addition, the network learning rate of bottom-up measurements is decreasing if the class prediction is correct. Since semantic neurons learn from episodic neurons' weight, it means that each semantic neuron encodes episodic events in a larger temporal size.

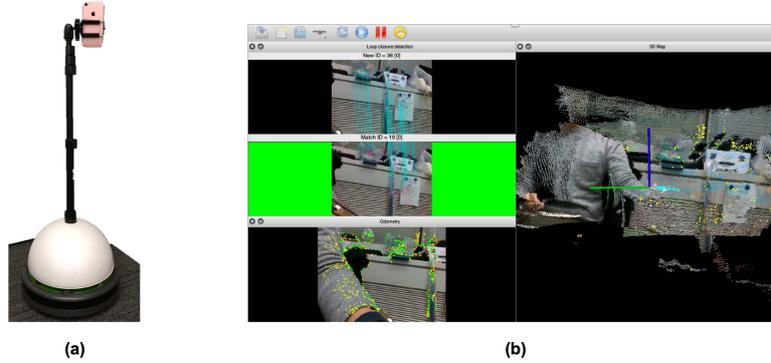


Fig. 2. (a) Robot attached with various sensors (b) Visual SLAM with surf features

Since episodic neurons will activate for a number $K^{\text{em}} + 1$ of incoming data, whereas each of semantic neurons will activate for a number $K^{\text{sm}} + 1$ of neurons activated in episodic memory layer. Therefore, the total input size that encoded in semantic neurons is $K^{\text{sm}} + K^{\text{em}} + 1$.

3 Experimental Setup & Results

We validate our proposed method using a iRobot Roomba robot that attached with a Hokuyo Laser scanner, Intel RealSense stereo camera, and Intel i5 processor computer as shown in Figure 2. The laser scanner signal was sampled at 10 Hz. Since the robot has to traverse the environment autonomously, we developed a Fuzzy motion movement behavior that allows the robot for obstacle avoiding and wall-following. The moving speed of the robot varies from 0.05m/s to 0.5m/s. The episodic memory layer receives a surf features vector with a temporal resolution of 3 scans ($K^{\text{em}} = 2$). As mentioned in section 2.3, the semantic memory layer will encode for a total of 5 scans in the experiment. For the semantic topological map building, classification and retrieval, we determine the predicted label of a new input data using the Equation 15 from semantic layer's best matching neuron. A place label is predicted for each incoming sensory information.

The experiments were conducted in the 7th floor of university corridor, study area, and rest area that connecting with each other. The grid map of the experimental place as shown in Figure 3(a). We conduct the experiment in such environmental conditions is to validate our proposed method is able to work in natural environment with moderate changing of environmental conditions.

We commanded the robot to traverse the experimental place starting from the study area and travel to the rest area through the corridor then back to the start point again. During the traverse, DM-ARM continually learns incoming sensory information and generates the semantic map for representing the environment. After each traverse, the robot will go for charging and then memory

10 Wei Hong Chin et al.

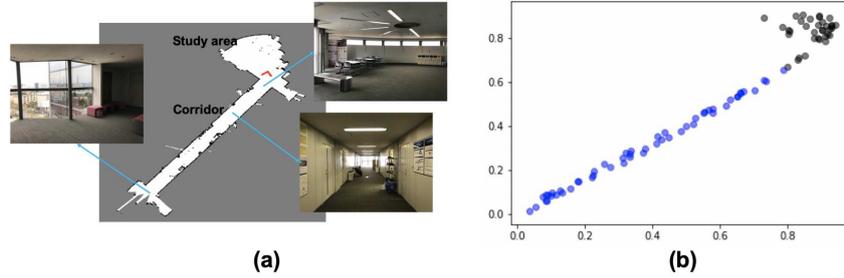


Fig. 3. (a) Experimental environment (b) Built Semantic Map

replay is triggered for semantic map memory consolidation. Once the robot is fully charged, memory replay is deactivated and the robot starts traversing and continuously learns and updates the semantic map again. For space labeling, we developed a speech-to-text iOS application to obtain the space labels for labeling the semantic map. As shown in Figure 3, the semantic map is segmented into 2 regions, the study area is colored in black and the corridor is colored in blue respectively.

After the first traverse, with the semantic information, the robot navigates the environment with different moving behaviors according to the region. For example, the robot is switched to obstacle avoidance mode in the study area where the place populated with moving people and furniture. When it traversed to the corridor, the moving behavior is then switched to the wall following mode and fast-speed mode since the corridor is a straight path.

We repeated the experiment for ten times and the generated spatial map quality was measured by Total Quantization Error (TQE) and localization rate. Figure 4 illustrates the total number of neurons in the map that generated by SOARIN and [11] for each traverse. TQE measures the similarity between sensory information and weights of episodic neurons in the spatial map. Figure 5 shows the TQE of the learning approach for each traverse. Next, the localization rate is measured by computing the euclidean distance between the winner neuron's encoded location and robot current location that obtained from the SLAM algorithm. The robot is localized successfully if the euclidean radius is within a threshold value ($0.01m$). The localization rate result is shown in Figure 6. Results showed that the proposed method able to generate a semantic topological map that can be used for path planning and robot moving behavior switching.

4 Conclusion

In this paper, we proposed a model termed as DM-ARM that models human declarative memory which can continually learn the spatiotemporal relationship of sensory data from both active sensors and proprioceptive indications to generate a semantic map incrementally. DM-ARM updates the semantic map by ex-

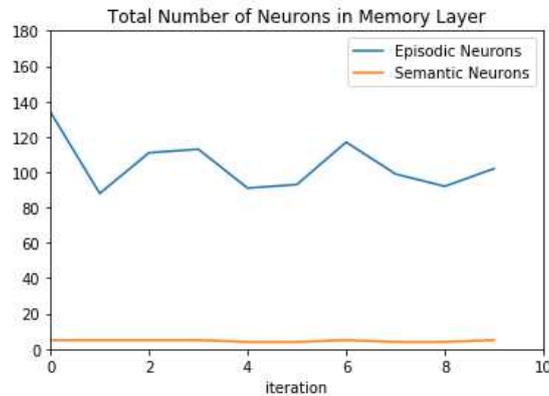


Fig. 4. Total number of generated neurons for each traverse

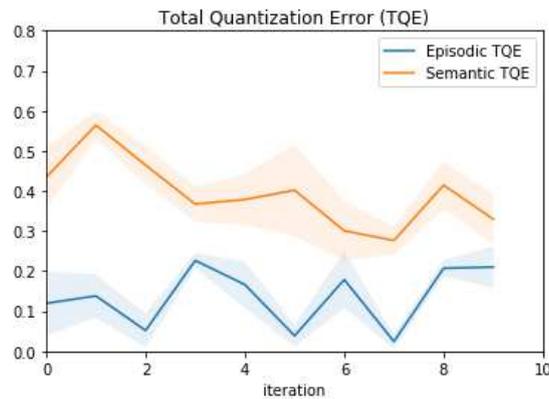


Fig. 5. Total Quantization Error of the learning method for each traverse

panding or shrinking its memory structure autonomously. In addition, DM-ARM consolidates the episodic semantic map through its self episodic memory playback without the needs of external sensory cues. DM-ARM has been validated through real robot implementation. In the future, we will integrate DM-ARM with path planning algorithm to utilize the topological structure of the spatial map for goal-directed navigation. Lastly, we will further improve and validate the performance of DM-ARM in more challenging and larger environments.

References

1. K. Rebai, O. Azouaoui and N. Achour: Bio-inspired visual memory for robot cognitive map building and scene recognition. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, 2012, pp. 2985-2990. (2012). doi:10.1109/IROS.2012.6385493

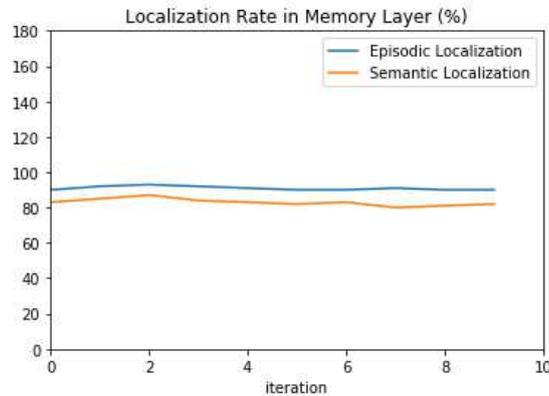


Fig. 6. Localization rate of the learning method for each traverse

2. Buzsáki, György and Moser, Edvard I.: Memory, navigation and theta rhythm in the hippocampal-entorhinal system. In: *Nature Neuroscience* 2013. vol. 16, pp. 130-138. Nature Publishing Group (2013). doi:doi.org/10.1038/nn.3304
3. J. Okeefe and L. Nadel: *The hippocampus as a cognitive map*. Clarendon Press Oxford, 1978, vol. 3 (1978).
4. B. L. McNaughton, F. P. Battaglia, O. Jensen, E. I. Moser, and M.-B. Moser: Path integration and the neural basis of the cognitive map. In: *Nature Reviews Neuroscience*. vol. 7, no. 8, pp. 663-678, (2006). doi:10.1038/nrn1932
5. E. Tulving: Episodic and semantic memory. In: *Organization of memory*, pp. 381-402, (1972).
6. B. Tian, V. A. Shim, M. Yuan, C. Srinivasan, H. Tang and H. Li: RGB-D based cognitive map building and navigation. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo*, pp. 1562-1567, (2013) doi:10.1109/IROS.2013.6696557
7. W. Yan, C. Weber and S. Wermter: A neural approach for robot navigation based on cognitive map learning. In: *2012 International Joint Conference on Neural Networks (IJCNN)*, Brisbane, QLD, pp. 1-8 (2012).
8. W. H. Chin, Y. Toda, N. Kubota, C. K. Loo and M. Seera: Episodic memory multimodal learning for robot sensorimotor map building and navigation. In: *IEEE Transactions on Cognitive and Developmental Systems*. doi:10.1109/TCDS.2018.2875309
9. Kumaran, Dharshan and Hassabis, Demis and McClelland, James L.: What Learning Systems do Intelligent Agents Need? Complementary Learning Systems Theory Updated. In: *Trends in Cognitive Sciences*. vol. 20, no. 7, pp. 512-534, (2016)
10. McClelland, J. L., McNaughton, B. L., and O'Reilly, R. C.: Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. In: *Psychological Review*. vol. 102 no.3, pp. 419-457 (1995). doi:10.1037/0033-295X.102.3.419
11. Parisi, G. I., Tani, J., Weber, C., and Wermter, S.: Lifelong learning of humans actions with deep neural network self-organization. In: *Neural Networks*. vol. 96, pp. 137-149 (2017). doi:10.1016/j.neunet.2017.09.001